

检索数据

一、关键字

作为SQL部分的保留字，不能用做表或者列的名字。

二、select

select检索表数据，必须至少给出两条信息：

- 想选择什么
- 从何处选择

2.1 检索单个列

```
select prod_name from products;
```

如果没有指定排序，则检索出来的数据默认是没有特定顺序的。可能是被添加时候的顺序，也不一定是。

分号: 多条SQL语句以分号分隔。多数DBMS不需要在单个SQL后面加分号，但也有的DBMS要求必须在单个SQL后面加分号。当然，如果愿意则可以一直加上分号，即使不需要，加上也没坏处。

大小写: SQL语句不区分大小写，因此SELECT 和select是等效的。

空格: SQL语句中，所有空格都被忽略，可以写成多行也可以写成一行。

2.2 检索多个列

SELECT后跟多个列，列之间用逗号分隔，最后一个列后面不加。如果最后一个列后追加了逗号，则会出现错误。

如：

```
select prod_id , prod_name , prod_price from products;
```

SQL语句一般返回原始的、无格式的数据。

2.3 检索所有列

除了指定所需的列，还可以检索所有列。在列名位置使用*通配符。

```
select * from products;
```

这样查出来的列的顺序一般是定义中出现的物理顺序，但并不总是如此。

除非你确实需要表中的每一列，否则最好不要使用*通配符，因为检索不必要的列通常会降低检索和应用程序的性能。

检索未知列：

使用通配符有一个优点，不明确指定列所以可以检索出名字未知的列。

2.4 检索不同的值

本身select返回所有匹配的行，但是如果你不希望每个值每次都出现，则可以使用**DISTINCT**,它指示数据库只返回不同的值。

```
select distinct vend_id from products;
```

告诉DBMS只返回不同的vend_id行，必须放在列名前面。

这个关键字作用于所有的列，不仅仅是跟在它后面那个列，比如：

```
select distinct vend_id ,prod_price
```

这两列不完全相同的时候，都会显示出来。

2.5 限制结果

select返回指定表中所有匹配的行，很可能是每一行。

如果你只想返回第一行或者一定数量的行，则可以根据数据库的不同使用不同的方式。

SQL Server使用top关键字：

```
select top 5 prod_name from products;
```

DB2使用ROWS ONLY：

```
select prod_name from products fetch first 5 rows only;
```

ORACLE使用rownum:

```
select prod_name from products where rownum <=5;
```

MySQL MariaDB PostgreSQL SQLite使用limit:

```
select prod_name from products limit 5;
```

limit 5表示返回的数据不超过5行。

假设只有9行数据，为了得到后面的数据，则需要指定从哪儿开始以及需要检索的行数。

```
select prod_name from products limit 5 offset 5;
```

表示返回第五行起的5行数据。第一个数字检索的行数，第二个数字从哪儿开始。

这个语句的结果则会返回4行数据，因为从第5行开始之后的最多5行数据，总共有9行数据，从第5编号开始一共4条数据。

0行:

第一个被检索的行是0行，而不是第一行。因此Limit 1 offset 1会检索第二行，而不是第一行。

MYSQL MARIADB SQLite的捷径:

他们支持简化版的limit 4 offset 3语句，即limit 3, 4。使用这个语法，逗号之前对应offset，逗号之后对应limit。

三、注释

SQL语句是由DBMS处理的指令，如果你希望看到不进行处理文本，也可以。

3.1 行内注释

```
select prod_name from products; --这是一条注释
```

使用--（两个连字符）嵌在行内，他们之后的文本就是注释。

```
#这也是行内注释  
select prod_name from products;
```

使用#也可以表示行内注释，#之后的一整行都代表注释。

3.2 多行注释

/*在这里放注释*/可以用来多行注释

```
/*这是一行注释  
这也是一行注释  
他们一起组成多行注释  
*/  
select prod_name from products;
```